



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/844,340	04/27/2001	Kevin Gary Tapperson	AUS920010051US1	9357

7590

09/08/2005

Duke W. Yee  
Carstens, Yee & Cahoon, LLP  
P.O. Box 802334  
Dallas, TX 75380

EXAMINER
----------

MEUCCI, MICHAEL D

ART UNIT	PAPER NUMBER
----------	--------------

2142

DATE MAILED: 09/08/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/844,340

Applicant(s)

TAPPERSON, KEVIN GARY

Examiner

Michael D. Meucci

Art Unit

2142

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --  
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 02 June 2005.
- 2a) ☐ This action is FINAL. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-40 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-40 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 06 August 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)  
Paper No(s)/Mail Date \_\_\_\_\_
- 4) ☐ Interview Summary (PTO-413)  
Paper No(s)/Mail Date. \_\_\_\_\_
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_

### DETAILED ACTION

1. This action is in response to arguments by applicant filed 02 June 2005.
2. Claims 1-40 currently pending.
3. Because new grounds of rejection are being made to substantially unamended claims, this action is **non-final**.

### *Response to Amendment*

4. Examiner acknowledges amendments made to the abstract to overcome objections.

### *Claim Rejections - 35 USC § 112*

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:  

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.
6. Claims 9-12, 16-17, 27-30, and 34-35 rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Each claim contains the trademark name "Java"

Claims 9-12, 16-17, 27-30, and 34-35 contain the trademark/trade name "Java". Where a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of 35 U.S.C. 112, second paragraph. See *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade

name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, a trademark or trade name does not identify or describe the goods associated with the trademark or trade name. In the present case, the trademark/trade name is used to identify/describe a Java virtual machine and, accordingly, the identification/description is indefinite.

Although the use of trademarks is permissible in patent applications, the proprietary nature of the marks should be respected and every effort made to prevent their use in any manner which might adversely affect their validity as trademarks. As such, the examiner recommends amendments to replace "Java Virtual Machine" in all instances with --Java bytecode on a virtual machine-- (see class 717, subclass 148 def: Just-in-time compiling or dynamic compiling (e.g., compiling Java bytecode on a virtual machine)).

### ***Claim Rejections - 35 USC § 103***

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

8. Claim 1-5, 7, 9-14, 16-17, 20-24, 26-32, 34-35, and 38-39 rejected under 35 U.S.C. 103(a) as being unpatentable over Endicott et al. (U.S. 6,047,295) hereinafter

referred to as Endicott in view of Howes et al. (U.S. 6,366,558 B1) hereinafter referred to as Howes and de la Salle (U.S. 5,878,420).

a. As per claims 1, 20, and 38, Endicott teaches: establishing a connection to a server (lines 33-37 of column 5); maintaining a normal reference to an object (lines 31-34 of column 1); and periodically destroying objects maintained by weak references (lines 24-33 of column 3).

Endicott does not explicitly teach: starting a timer responsive to conclusion of a communication process using the connection; and responsive to conclusion of a predetermined time period measured by the timer, maintaining a weak reference to the connection object. However, Howes discloses: "The timer field is used to time out a connection object when no activity occurs on the connection for a specified period of time," (lines 10-12 of column 14). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to start a timer responsive to conclusion of a communication process using the connection; and responsive to conclusion of a predetermined time period measured by the timer, maintain a weak reference to the connection object. "While the standby Local Director is inactive, it does not handle packets for connections and so none of the connection object timers in the standby Local Director are updated when connection activity occurs. Thus, many of the connection objects on the standby Local Director would be timed out. By updating the timer field to the current time when the standby Local Director becomes active, the connection objects are all preserved and are not timed out until a time out period expires beginning from the time that the standby Local Director becomes active," (lines

12-23 of column 14 in Howes). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to start a timer responsive to conclusion of a communication process using the connection; and responsive to conclusion of a predetermined time period measured by the timer, maintain a weak reference to the connection object in the system as taught by Endicott.

Endicott does not explicitly teach: a connection object. However, de la Salle discloses: "In order to create the connection object 80, the transport decode routine 90 utilizes a hash table based on a pair of matched endpoint addresses appearing in the same captured packet 22'. These can either be a pair of network addresses 23 or a pair of board addresses 36'. The connection object 80 is an identified and separately "named" (numbered) object representing a link between two defined endpoints. For example, a particular captured packet 22' may involve a print spool request from Joan's PC (24) to a printer control server (32), designating an item for printing on an associated dot matrix printer. A connection object 80 would then be created and identified ("named") which would indicate a "connection" between Joan's PC and the printer server," (lines 28-50 of column 8). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a connection object as the object. "A later captured packet 22' could include a further print request traveling along the same path 18 as before from Joan's PC to the printer server, with either the same designated printer, or, say, a laser printer. In this case a separately named connection object 80, with essentially identical content, other than the name, would be created to memorialize a second "connection" between the same two endpoints. Sorting and

counting of these objects is performed at the level of the database builder aspects of the analysis assembly, so each of these separate objects is simply stored in the buffer 44 until called for," (lines 50-61 of column 8 in de la Salle). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to utilize a connection object as the object in the system as taught by Endicott and Howes.

b. As per claims 2 and 21, Endicott teaches: determining whether a normal reference to the connection object exists (lines 43-52 of column 7); and reusing the connection if the normal reference exists (lines 35-47 of column 10).

c. As per claims 3 and 22, Endicott does not explicitly teach: restarting the timer. However, Howes discloses: "In a step 930, the timer field is updated in all of the connection objects on the newly active Local Director. The timer field is used to time out a connection object when no activity occurs on the connection for a specified period of time (lines 8-12 of column 14). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to restart the timer. "By updating the timer field to the current time when the standby Local Director becomes active, the connection objects are all preserved and are not timed out until a time out period expires beginning from the time that the standby Local Director becomes active," (lines 18-23 of column 14 in Howes). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to reset the timer in the system as taught by Endicott.

d. As per claims 4, and 23, Endicott teaches: determining whether a weak reference to the connection exists (lines 30-39 of column 2, lines 43-47 of column 7, and line 48 of column 10 through line 2 of column 11); determining whether the connection object has been destroyed if the weak reference exists (lines 47-52 of column 7 and lines 44-47 of column 14); reusing the connection if the connection object has not been destroyed (line 48 of column 10 through line 2 of column 11 and lines 33-50 of column 14).

e. As per claims 5 and 24, Endicott does not explicitly teach: restarting the timer. However, Howes discloses: "In a step 930, the timer field is updated in all of the connection objects on the newly active Local Director. The timer field is used to time out a connection object when no activity occurs on the connection for a specified period of time (lines 8-12 of column 14). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to restart the timer. "By updating the timer field to the current time when the standby Local Director becomes active, the connection objects are all preserved and are not timed out until a time out period expires beginning from the time that the standby Local Director becomes active," (lines 18-23 of column 14 in Howes). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to reset the timer in the system as taught by Endicott.

f. As per claims 7 and 26, Endicott teaches: sending notification to the server that the connection object is unreferenced when a weak reference to the connection object is maintained (line 66 of column 6 through line 15 of column 7).

g. As per claims 9-10 and 27-28, Endicott teaches: the client is a client Java Virtual Machine; the server is a server Java Virtual Machine (lines 36-42 and lines 54-62 of column 6).

h. As per claims 11-12 and 29-30, Endicott teaches: the client Java Virtual Machine and the server Java Virtual Machine reside on the same host machine (lines 30-45 of column 5, lines 36-42 and lines 54-62 of column 6).

i. Claims 13, 31, and 39 contain similar limitations as claims 4 and 23 above and are rejected under the same rationale.

j. As per claims 14 and 32, Endicott teaches: destroying the connection object responsive to garbage collection by the server (lines 46-67 of column 3).

k. As per claims 16 and 34, Endicott teaches: the connection object is a Java object (lines 36-42 of column 6).

l. As per claims 17 and 35, Endicott teaches: the Java object is a remote method invocation object (lines 56-59 of column 5).

9. Claims 6, 15, 25, and 33 rejected under 35 U.S.C. 103(a) as being unpatentable over Endicott, Howes, and de la Salle as applied to claims 4 and 23 respectively above in view of Official Notice.

Official Notice taken of establishing a new connection if the connection object has been destroyed. A connection object is implicitly *new* if the previous connection object was destroyed. Establishing a new connection if the connection object has been destroyed is very well known in the art at the time of the applicant's invention. It would

have been obvious for one of ordinary skill in the art at the time of the applicant's invention to establish a new connection if the connection object has been destroyed in the system as taught by Endicott.

10. Claim 8 rejected under 35 U.S.C. 103(a) as being unpatentable over Endicott, Howes, and de la Salle as applied to claim 1, in view of Wollrath et al. (U.S. 5,832,529) hereinafter referred to as Wollrath.

As per claim 8, Endicott teaches: destroying the connection object in response to garbage collection (lines 1-4 of column 2 and lines 46-67 of column 3). Endicott does not explicitly teach: garbage collection is done by a server. However, Wollrath discloses: "The server call processor also initiates a garbage collection cycle," (lines 20-21 of column 7). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to have a server do the garbage collection. "The server call processor also initiates a garbage collection cycle to reclaim resources for which it determines that either no more references are being made to the resource or that the agreed lease period for the resource has expired," (lines 20-23 of column 7 in Wollrath). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to have a server do the garbage collection in the system as taught by Endicott.

11. Claims 18, 36, and 40 rejected under 35 U.S.C. 103(a) as being unpatentable over Endicott, Howes, and de la Salle in view of Geise et al. (U.S. 5,247,520) hereinafter referred to as Geise.

As per claims 18, 36, and 40, Endicott teaches: a hash map (lines 56-59 of column 2) and removing the reference to the connection object from the hash map (lines 4-9 of column 3).

Endicott does not explicitly to teach: starting a timer responsive to conclusion of a communication process using the connection. However, Howes discloses: "The timer field is used to time out a connection object when no activity occurs on the connection for a specified period of time," (lines 10-12 of column 14). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to start a timer responsive to conclusion of a communication process using the connection. "While the standby Local Director is inactive, it does not handle packets for connections and so none of the connection object timers in the standby Local Director are updated when connection activity occurs. Thus, many of the connection objects on the standby Local Director would be timed out. By updating the timer field to the current time when the standby Local Director becomes active, the connection objects are all preserved and are not timed out until a time out period expires beginning from the time that the standby Local Director becomes active," (lines 12-23 of column 14 in Howes). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to start a timer responsive to conclusion of a communication process using the connection in the system as taught by Endicott.

Endicott does not explicitly teach: removing the reference responsive to conclusion of a predetermined time period measured by the timer. However, Howes discloses: "In one embodiment, connections are timed out periodically and deleted when the recorded time in timer 362 differs from the current time by more than a determined amount," (lines 24-27 of column 9). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to remove the reference responsive to conclusion of a predetermined time period measured by the timer. "The timer field is used to time out a connection object when no activity occurs on the connection for a specified period of time. While the standby Local Director is inactive, it does not handle packets for connections and so none of the connection object timers in the standby Local Director are updated when connection activity occurs. Thus, many of the connection objects on the standby Local Director would be timed out," (lines 10-18 of column 14 in Howes). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to remove the reference responsive to conclusion of a predetermined time period measured by the timer in the system as taught by Endicott.

Endicott does not explicitly teach: adding a reference to a connection object for a connection to a weak hash map and a hash map. However, Geise discloses: "LCB 62 also contains pointers to two hash tables 76 and 78. Table 76 is an ALS.sub.-- ID hash table, and table 78 is a CONNECTION.sub.-- ID hash table," (lines 64-66 of column 5). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to add a reference to a connection object for a connection to a

weak hash map and a hash map. "Each valid entry in both has tables 76, 78 contains a pointer to one VCCB (dashed lines). Each hash table 76, 78 must have at least as many entries as the maximum number of logical links which can be handled simultaneously over the physical link represented by LCB 62," (line 66 of column 5 through line 3 of column 6 and Fig. 5). It is for this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to add a reference to a connection object for a connection to a weak hash map and a hash map in the system as taught by Endicott.

12. Claims 19 and 37 rejected under 35 U.S.C. 103(a) as being unpatentable over Endicott, Howes, de la Salle, and Geise as applied to claims 18 and 36 respectively above, further in view of Weinstein et al. (Google Groups comp.lang.java.databases) hereinafter referred to as Weinstein.

As per claims 19 and 37, Endicott teaches: determining whether the connection object has been destroyed (lines 47-52 of column 7 and lines 44-47 of column 14); removing the reference to the connection object from the weak hash map if the connection object has been destroyed (lines 56-59 of column 2, lines 24-33 of column 3, and lines 33-50 of column 14).

Endicott fails to teach: maintaining the reference to the connection object if the connection object has not been destroyed to thereby allow use of such connection object by a subsequent communication process between the client and server without establishing a new connection between the client and server. However, Weinstein

discloses: "From that driver, the client obtains a JDBC connection object which communicates (only) with the WebLogic Application Server already running in another JVM. This server may already be handling other clients in the same or a different way. This server will be running after this an(d) any other clients have finished and are gone," and "It can either provide JDBC connection pools, or retain a client-specific DBMS connection so a client can retain transactional state during repeated log-ins/outs," (both from page 12 of 16). It would have been obvious to one of ordinary skill in the art at the time of the applicant's invention to maintain the reference to the connection object if the connection object has not been destroyed to thereby allow use of such connection object by a subsequent communication process between the client and server without establishing a new connection between the client and server. "All this is configurable/alterable by the client using WebLogic extensions to JDBC. The Application Server connection pools can allow JDBC access to 3-tier clients who needn't transmit, know, or find out the DBMS password, or location. In 4.0, client code can create, manage, disable, re-enable, and destroy pools in the middle tier, dynamically. The middle tier and 3-tier driver can communicate via several client-selectable protocols, including secure protocols that can cross third-party firewalls. The middle tier also provides integrated security for all this.

There we are. Three well defined and physically distinct, independent, separately running tiers. Each providing specific business value added to the JDBC proposition, and WebLogic extensions to JDBC to allow JDBC/T3 clients to configure the behavior of the middle tier with regards to JDBC for the client," (page 12 of 16 in Weinstein). It is for

this reason that one of ordinary skill in the art at the time of the applicant's invention would have been motivated to maintain the reference to the connection object if the connection object has not been destroyed to thereby allow use of such connection object by a subsequent communication process between the client and server without establishing a new connection between the client and server in the system as taught by Endicott, Howes, and Geise.

### ***Response to Arguments***

13. (A) Applicant contends that Endicott does not teach use of a connection object with respect to claim 1. This argument is moot in view of new grounds of rejection, but is further explained below.

As to point (A), although not explicitly disclosed in Endicott, use of a "connection" object was extremely obvious at the time of the applicant's invention. The objects of Endicott are generic objects, so nearly any implementation and use of them can be construed. Examiner urges applicant to research "session" relating to computer science in some manner. Wikipedia ([www.wikipedia.org](http://www.wikipedia.org)) defines a session as: "In computer science, in particular networking, a **session** is either a lasting connection using the session layer of a network protocol or a lasting connection between a user (or user agent) and a peer, typically a server, usually involving the exchange of many packets between the user's computer and the server. A session is typically implemented as a layer in a network protocol (e.g., telnet or FTP). In the case of transport protocols which do not implement a formal session layer (e.g., UDP) or where sessions at the session

layer are generally very short-lived (e.g., HTTP), sessions are maintained by a higher level program using a method defined in the data being exchanged. For example, an HTTP exchange between a browser and a remote host may include an HTTP cookie which identifies state, such as a unique session ID, information about the user's preferences or authorisation level, etc." As such, use of sessions in combination with a Java Virtual Machine is not novel and obvious over the prior art. Also, the obvious addition of the de la Salle reference overcomes any and all claims by applicant, that the instant application is patentable over the prior art, when arguing that Endicott does not teach a "connection object". These arguments have not been addressed any further.

14. (B) Applicant contends that Wolczko fails to teach the claimed steps of *responsive to conclusion of a communication process using the connection, starting a timer and responsive to conclusion of a predetermined time period measured by the timer, maintaining a weak reference to the connection object* with respect to claim 1. These arguments are moot in view of new grounds of rejection.

15. (C) With respect to claim 4, the applicant contends that Endicott does not teach (i) *determining whether a weak reference to the connection object exists*; (ii) *determining whether the connection object has been destroyed if the weak reference exists*; and (iii) *reusing the connection if the connection object has not been destroyed*. The examiner respectfully disagrees.

As to point (C), the applicant argues that the cited passage merely describes the existence of a flag associated with a weak reference. Limitations (i) and (ii) are implicit in the disclosed garbage collector of the virtual machine (see citations of column 7 and additionally page 2 of the specification of the instant application in the background section). As to the arguments that step (iii) is a conditional action directed to reusing a connection if the connection object has not been destroyed, the examiner points out that steps 106 and 108 of Fig. 5 show the weak reference being returned for use. The description of this is clearly disclosed on line 60 of column 10 through line 2 of column 11.

16. (D) With respect to claim 7, the applicant contends that none of the cited references teach *sending notification to the server that the connection object is unreferenced when a weak reference to the connection object is maintained*. The examiner respectfully disagrees.

As to point (D); the applicant argues that the passage describes management of weak object references, and describes a technique of selectively inhibiting access by other program threads to only those weak references that reference objects not yet known to be strongly-reachable during a garbage collection cycle. Since the issue of "connection objects" has already been discussed above in point (A), so clients and servers are implicitly present in the system. It is also implicit that a notification is sent to the server if a weak reference to the connection object is maintained. The connection is two-way, and therefore the server receives some information regarding whether or not

the connection is in an idle state and available for use or "unreferenced" except by the weak reference.

17. (E) With respect to claim 8, the applicant contends that none of the cited references teach *wherein the step of periodically destroying connection objects maintained by weak references comprises destroying the connection object in response to garbage collection by the server*. This argument is moot in view of new grounds of rejection.

18. (F) With respect to claim 13, see points (A) and (C) above.

19. (G) With respect to claims 6, 15, 25, and 33, the applicant contends that none of the cited references teach or suggest the claimed feature of *establishing a new connection if the connection object has been destroyed*. The examiner contends that no additional reference is necessary since this limitation is implicit in any system attempting to establish a new connection. Once a connection object is destroyed, the connection no longer exists. Thus, a new connection must be established to regain functionality of this system and any system. Examiner provides Gudjonsson et al. (U.S. 6,564,261 B1) as one of many references supporting evidence towards the *blatant* obviousness of this limitation (see lines 61 of column 30 through line 21 of column 31).

20. (H) With respect to claims 18, 36, and 40, the applicant contends that none of the cited references teach or suggest (i) *adding a reference to a connection object for a connection to a weak hash map and a hash map*; (ii) *responsive to conclusion of a communication process using the connect, starting a timer, and* (iii) *responsive to conclusion of a predetermined time period measure by the timer, removing the reference to the connection object from the hash map*. The examiner respectfully disagrees.

As to point (H), the examiner contends that step (i) is admitted prior art (see lines 14-23 of page 14 in the specification). As such, a weak hash map is not functionally different from a hash map and only differs in what the system decides to store in it. Therefore, Endicott teaches a weak hash map as previously cited. Storing an object in multiple tables (or hash maps in this case) is not new or novel over the prior art either. If an object can be stored in a single table or hash map, it can just as easily be stored in a second, third, etc. Examiner provides Gibbons et al. (U.S. 5,761,511) as supporting evidence which discloses shadow maps.

See point (B) regarding steps (ii) and (iii).

21. (I) With respect to claims 19 and 37, the applicant contends that none of the cited references teach or suggest *maintaining the reference to the connection object in the weak hash map if the connection object has not been destroyed to thereby allow the use of such connection object by a subsequent communication process between the*

*client and server without establishing a new connection between the client and server.*

The examiner respectfully disagrees.

As to point (I), the applicant argues that the cited reference is not enabling in that it does not teach a weak hash map. The examiner contends that Weinstein reference is directed towards maintaining the reference to the connection object if the connection object has not been destroyed to thereby allow use of such connection object by a subsequent communication process between the client and server without establishing a new connection between the client and server. The examiner contends that hash maps and weak hash maps have been cited in other references. The mere discussion of reusing a connection object in Weinstein clearly shows that this concept is very well known in the art. Further evidence of this can be seen in Quinlan (U.S. 6,338,089 B1) and Tavs et al. (U.S. 6,073,175) which both disclose reusing existing session connections. Also, a recitation of the intended use of the claimed invention must result in a structural difference between the claimed invention and the prior art in order to patentably distinguish the claimed invention from the prior art. If the prior art structure is capable of performing the intended use, then it meets the claim.

### ***Remarks***

22. The examiner believes that the applicant's invention is directed more towards the reusing of weakly referenced connection objects in the weak hash maps than towards the *broadly* claimed subject matter disclosed in the independent claims. To further prosecution, the examiner requests that the applicant *substantially* amend the claims,

Art Unit: 2142

remembering to not include new matter, focusing on the applicant's invention as a whole, rather than segmenting the claims as apparent distinct processes.

### ***Conclusion***

23. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Jackson et al. (U.S. 5,274,804) discloses garbage collection and memory reclamation.

Topka et al. (U.S. 5,398,334) discloses automatic garbage collection using strong and weak encapsulated pointers.

Ding et al. (U.S. 5,699,361) discloses garbage collection using a timer.

Gibbons et al. (U.S. 5,761,511) discloses shadow maps.

Tavs et al. (U.S. 6,073,175) discloses reuse of open connections from a user session.

Endicott et al. (U.S. 6,098,080) discloses garbage collection.

Quinlan (U.S. 6,338,089 B1) discloses reusing session connections.

Kirk, III (U.S. 6,421,690 B1) discloses memory management and garbage collection.

Hughes (U.S. 6,429,860 B1) discloses use of the Java virtual machine.

Hudson et al. (U.S. 6,671,707 B1) discloses concurrent copying garbage collection.

Burton et al. (U.S. 6,874,074 B1) discloses memory reclamation

24. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Michael Meucci at (571) 272-3892. The examiner can normally be reached on Monday-Friday from 9:00 AM to 6:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Andrew Caldwell, can be reached at (571) 272-3868. The fax phone number for this Group is 571-273-8300.

Communications via Internet e-mail regarding this application, other than those under 35 U.S.C. 132 or which otherwise require a signature, may be used by the applicant and should be addressed to [michael.meucci@uspto.gov].

All Internet e-mail communications will be made of record in the application file. PTO employees do not engage in Internet communications where there exists a

Art Unit: 2142

possibility that sensitive information could be identified or exchanged unless the record includes a properly signed express waiver of the confidentiality requirements of 35 U.S.C. 122. This is more clearly set forth in the Interim Internet Usage Policy published in the Official Gazette of the Patent and Trademark on February 25, 1997 at 1195 OG 89.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

  
BEATRIZ PRIETO  
PRIMARY EXAMINER